Universidad Simón Bolívar
Departamento de Computación y Tecnología de la Información
CI-56512- Diseño de Algoritmos I
Trimestre Abril-Julio 2012

Proyecto1: Resolviendo el *Symmetric Traveling Salesman Problem*

1. Planteamiento del Problema

Se desea que implemente un software que encuentre soluciones de alta calidad y con un esfuerzo computacional adecuado, para el *Symmetric Traveling Salesman Problem*. La información sobre el *Symmetric Traveling Salesman Problem* (STSP) la puede obtener del libro *The traveling salesman problem and its variations* de Gutin y Punnen [2]. Se quiere resuelva este problema implementando un algoritmo heurístico que esté basado en alguno de estos tres métodos: *Iterated Hill Climber*, *Stochastic hill-climber*[4] ó *Simulated annealing*. La implementación de su programa debe hacerse usando alguno de estos tres lenguajes de programación: C, C++ ó JAVA.

Para probar su algoritmo debe hacer uso de cuatro instancias bien conocidas del STSP que son: *eil51*, *kroA100*, *d198* y *rat783*. Estas instancias y su valor óptimo se encuentran en la página web TSPLIB [5], la cual es depósito de instancias e información del *Traveling Salesman Problem*. Su programa debe recibir, como una de sus entradas, el archivo con la instancia, tal como se encuentra en TSPLIB sin modificar su formato. Al finalizar la ejecución su programa debe mostrar por la salida estándar cuatro elementos: el orden en que son visitadas las ciudades (*tour*), la distancia del recorrido, el número de iteraciones del algoritmo que resuelve el STSP y el tiempo de ejecución del algoritmo en segundos. También puede mostrar por la salida estándar cualquier otra información que considere relevante.

Además del programa que resuelve el *Symmetric Traveling Salesman Problem*, debe realizar un informe que contenga los siguientes secciones:

- *Portada*
- *Introducción:*
  1. Motivación del proyecto.
  2. Breve descripción del problema.
  3. Descripción del contenido del informe.
- *Algoritmo para el STSP:* Debe describir de manera clara y precisa el algoritmo que diseño para resolver el STSP. La idea es que cualquier profesional competente en ciencias de la computación que lea esta sección, debe ser capaz de implementar la heurística que usted desarrolló y de reproducir los resultados que usted obtuvo. Debido a esto la sección debería contener por lo menos:
  1. Las estructuras de datos usadas para representar el problema y otras que considere importantes
  2. La descripción de los principales algoritmos para resolver el problema. Preferiblemente debe mostrar el pseudocódigo de esos algoritmos
  3. Los parámetros que usa su algoritmo y la justificación de los mismos. Por ejemplo la condición de parada, el valor de la temperatura en el caso de usar *Simulated annealing*, etc.
  4. Cualquier otra información que usted considere relevante para la implementación del algoritmo.

  Debe justificar y explicar el diseño de su solución.

- *Instrucciones de operación:* Descripción detallada de como compilar y correr su aplicación, así como el estado actual de la misma
- *Resultados Experimentales y Discusión:* (Ver indicaciones más adelante)
- *Conclusiones y Recomendaciones*
- *Referencias bibliográficas*

En cuanto a la sección de *Resultados Experimentales y Discusión* se desea hacer un estudio experimental que permita caracterizar el rendimiento de la solución algorítmica propuesta por usted. Según Barr et al. [1] la evaluación experimental de una heurística debería tomar en cuenta tres aspectos principales.

- *Calidad de la soluciones* que son obtenidas. El objetivo de este punto es ver qué tan cercanos son los valores obtenidos de los valores óptimos.
- *Esfuerzo computacional.* El tiempo de cómputo que debe ser invertido para obtener soluciones factibles de alta calidad.
- *Robustez y fiabilidad.* Es la habilidad de la heurística de obtener buenos resultados en una amplia gama de problemas y sin necesidad de tener que realizar demasiados ajustes en la configuración de los parámetros del algoritmo.

En este proyecto se debe evaluar el método heurístico implementado bajo la óptica de los tres puntos expuestos anteriormente. Debe presentar dos tablas de resultados experimentales. La primera tabla debe tener los siguientes datos:

- Nombre de la instancia
- Distancia promedio de 10 corridas de la heurística
- Porcentaje de desviación de la distancia promedio de la heurística, con respecto a la solución óptima
- Desviación estándar del valor promedio de la heurística
- Distancia de la mejor solución obtenida en las 10 corridas de la heurística
- Número de ocurrencias de la mejor solución en las 10 corridas de la heurística

La segunda tabla debe mostrar los siguentes resultados:

- Nombre de la instancia
- Distancia de la solución óptima
- Distancia de la mejor solución obtenida en las 10 corridas de la heurística
- Porcentaje de desviación de la mejor solución de la heurística con respecto a la solución óptima
- Tiempo promedio de las 10 corridas de la heurísticas, en segundos.

También puede incluir cualquier otra tabla o gráfico que considere relevante. El porcentaje de desviación de una solución de la heurística, con respecto a la solución óptima se calcula con la siguiente fórmula: $\frac{distanciaHeur - distanciaOpt}{distanciaOpt} * 100$. Las pruebas las puede realizar en cualquier computador. En el informe debe indicar las características del mismo: el modelo de procesador, la velocidad del reloj del procesador, la memoria RAM del sistema y el sistema de operación instalado. Su aplicación debe poder instalarse y ejecutarse en un equipo del LDC.

Se les proporcionará de implementaciones de los algoritmos de búsqueda local para el STSP, 2-opt y 3-opt [3], que puede usar en su software si así lo requiere.

2. Sobre la entrega

Este proyecto es individual y tiene un valor de 20 % de la nota final. La fecha límite de entrega es el día viernes 18 de mayo de 2012 hasta las 11:30 pm. Debe entregar el informe del proyecto y un archivo **.tar.gz** con el código del proyecto.

3. Consideraciones Finales

- Cualquier error que sea hallado en este enunciado, así como cualquier tipo de observación adicional sobre el proyecto, serán publicadas como fe de erratas en la página web del curso. Es responsabilidad de los alumnos revisar periódicamente la misma

- No debe haber copia, ni intercambio de información específica ni ayuda detallada entre los alumnos del curso. El incurrir en cualquiera de las acciones descritas anteriormente tendrá como consecuencia sanciones severas

Referencias

[1] Barr, R., Golden, B., Kelly, J., Resende, M., and Stewart, W. Designing and reporting on computational experiments with heuristic methods. *Journal of Heuristics 1*, 1 (1995), 9–32.

[2] Gutin, G., and Punnen, A. *The traveling salesman problem and its variations*, vol. 12. Kluwer Academic Pub, 2002.

[3] Hoos, H., and Stützle, T. *Stochastic local search: Foundations and applications.* Morgan Kaufmann, 2005.

[4] Michalewicz, Z., and Fogel, D. *How to solve it: modern heuristics.* Springer-Verlag New York Inc, 2000.

[5] Reinelt, G. Tsplib. `http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/`, 2012.